

BENCHMARKS

VOLUME 3 NUMBER 7

NORTH TEXAS STATE UNIVERSITY

SEPT/OCT 1982

*BENCHMARKS was produced using Waterloo/SCRIPT

TABLE OF CONTENTS

1.	BENCHMARKS HAS NEW FORMAT	2
2.	CONSULTING SERVICES AVAILABLE TO STUDENT PROGRAMMERS. . .	2
3.	CPM COMPOSITION AVAILABLE THROUGH THE UNIVERSITY PRINT SHOP.	3
4.	KNOW YOUR REPRESENTATIVE	4
5.	USING THE COMPUTER FOR RESEARCH: PART VI.	6
6.	SPSS.	10
	Doing Repeated Measures Analysis in MANOVA	10
7.	BMDP.	11
	BMDP Installed	12
8.	SAS	16
	New Release of SAS Installed	16
	New Features of SAS 79.6	17
	PROC COMPUTAB: SAS's Answer to VISICALC.	23
9.	MUSIC	24
	STATPAK Manuals Available.	24
	ASCII Terminals Under VM	24
	ID-Code Renewal.	25
	Disk Space Limited!.	25
	Microcomputer Software on MUSIC.	25
	MUSIC Usage Statistics 1981-1982	28
10.	HP-2000	29
	HP Notes	29
	HP 2000 Usage Statistics for '81-'82 Fiscal Year	29
11.	NTSU COMPUTING CENTER GENERAL USAGE EQUIPMENT LIST. . . .	30
12.	SOFTWARE - WHAT WE HAVE AND WHAT WE SUPPORT	32
13.	COMPUTER HUMOR.	36
	The COBOL Programmer's Prayer.	36
14.	INDEX TO PAST ISSUES.	37

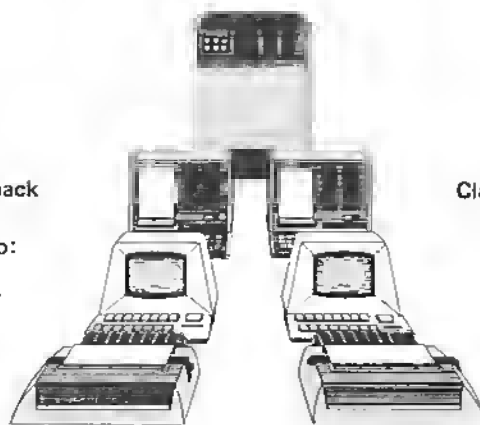
BENCHMARKS Reader/User feedback is encouraged.

Send all letters, suggestions, etc., to:
North Texas State University
The Computing Center
NT Station, Box 13495
Denton, Texas 76203

Claudia Putnam, BENCHMARKS Editor

Richard Harris,
Director of Computer Systems

Thomas Madron, Manager,
Academic Computing Services



Services Available To Users Of The NTSU Computing Facilities

All people mentioned below may be contacted by calling
(817) 565-2324:

Information and Project Numbers- Carolyn Goodman in the
Computing Center reception Area, ISB 119.

Test Scoring and Analysis - Carolyn Goodman.

Newsletter Questions/Contributions/etc. - Claudia Putnam.

Statistical/Research Support (provided for graduate students
and faculty members) - Bob Brookshire, George Morrow,
Claudia Putnam, and Mohamad Salahshoor.

Non-Research Student Programming Problems - student consultants
from the Computer Science Department, found in ISB 134A
near dispatch and the user keypunch area. Student
consulting provided by the College of Business is
available at the BA Computing Access Facility.

JCL and Debugging Problems - Mohamad Salahshoor.

Pre-Research Counseling, Including Survey Instrument Design -
Bob Brookshire, George Morrow, and Claudia Putnam.

Data Entry to MUSIC, Keypunch Requests and Questions Regarding
Layout of Keypunch Sheets; Interpreting - Betty Grise,
ISB 227.

Academic Timesharing Information and/or Problems HP/2000 and
AS/5000 MUSIC (McGill University System for Interactive
Computing) information and problems, including terminal
problems - Mohamad Salahshoor.

Administrative Applications - Coy Hoggard.

AS/5000 Computer Hardware/Software/Billing Problems - Sandy
Franklin.

JOB Submission and Retrieval - RJE Operators.

Fall Computing Hours

Computing facilities will be open during the following times
throughout the Fall semester (not applicable to holidays):

Computing Center RJE: 7 AM-4 AM, Monday-Friday; 8 AM-MN
Saturday; Noon-MN Sunday.

College of Business RJE: 8 AM-MN, Monday-Saturday; Noon-MN
Sunday.

Media Library (GAB): 8 AM-MN, Monday-Thursday; 8 AM-6 PM
Friday; 11 AM-5 PM Saturday; 4 PM-10 PM
Sunday.

BENCHMARKS HAS NEW FORMAT

Whether you have noticed it or not, BENCHMARKS has changed its looks a bit during the past several months. This is because the text processing package used to format this newsletter has been changed from MUSIC/SCRIPT to Waterloo/SCRIPT. Some of the advantages of Waterloo/SCRIPT over MUSIC/SCRIPT include the ability to submit your output directly to the high-speed printer in the Computing Center, the ability to use bold face type, and the ability to enclose things in boxes. For more information on Waterloo/SCRIPT, refer to the article "Waterloo SCRIPT Available in Batch Mode" in BENCHMARKS, Volume 2, Number 7. (NOTE: The program to print manuals described in the article should be SCRIPTMN instead of SCRIPTM.)

CONSULTING SERVICES AVAILABLE TO STUDENT PROGRAMMERS

The Computing Center offers research and statistical programming consultation to graduate students and faculty members, and will provide JCL and debugging help on a limited basis to anyone who needs it. The type of help often needed by Computer and Information Science undergraduate students, however, must be sought elsewhere.

The Computer Science Department hires student consultants who are on duty in ISB 134B approximately 15 hours each week beginning the 4th week of the semester. Their primary function is to help students in CSCI classes 110, 111, 232, 201, 340 and 370 with program design and syntax errors in BASIC, PASCAL, FORTRAN, 360 Assembly, SNOBOL and PL/I. They do, however, insofar as they are able, consult with any student from any department. When there are people waiting, consulting time is limited to 5-10 minutes per student. The consulting hours this semester are:

Monday	5 PM to 8 PM
Tuesday	10 AM to 1 PM
Wednesday	3 PM to 6 PM
Thursday	1 PM to 4 PM
Saturday	2 PM to 5 PM

No consultants are scheduled during final week. Any questions regarding the Computer Science consultants should be directed to Susan Rulon, Microcomputer Lab Director, GAB 330A, ext. 2810.

The College of Business computing facility is divided into three areas, the Microcomputer Lab, the Minicomputer Lab, and the area that is operated by the Computing Center (the BA RJE). According to Cengiz Capan, Manager of the Business Departmental Computing Center, there is at least one student consultant stationed in each area at all times. Their duties in order of priority are:

1. To monitor the area, schedule terminal time, etc.
2. To help students with problems such as logging-on, setting terminals, using modems, etc.

3. To consult with students on programming problems. This service has the lowest priority, and will only be offered if the consultant has completed the above mentioned duties. An exception to this is in the Microcomputer Lab, where there are two consultants, one to help with programming problems, the other to take care of the "housekeeping."

The Operations section of the Computing Center provides the following services at its Remote Job Entry Stations, located in the Information Sciences Building (room 134) and the Business Administration Building (room 152). Services marked with an asterisk (*) are available only at the ISB RJE.

1. Distribution of output routed to RJE printers. *Punched output distributed.
2. Information on individual job processing status.
3. Job submission: instructing new users in operation of the RJE card readers, and ascertaining validity of the JOB card. *Reading in card decks for users when the system has been down, or the card reader is inoperative.
4. Basic guidance in the use of the keypunch machines and program drums. *Assisting users in logging onto a display or hard-copy terminal.
5. Running 80/80 listings and *performing card utilities.
6. Sample JCL setups for the most frequently used languages, and *manuals for all current languages and statistical packages.
7. *Special handling requests for jobs, such as tape input, tape output, punched output, or special forms can be phoned in at 565-3860.
8. *Tapes are available for rental on a monthly basis. Users may copy their private tapes and create new data sets on these rental tapes, and the tapes remain in the center for use when specified in processing the user's job.
9. *Privately owned tapes may currently be checked in and out of the Academic tape library by users, which provides another in-center area to keep tapes that will be used in processing your job. (This service will be discontinued by January 1983, when an on-line tape management system will be instituted. In January, all privately owned tapes must be copied onto Computing Center rental tapes to be used in job processing).

CPM COMPOSITION AVAILABLE THROUGH THE UNIVERSITY PRINT SHOP

The University Print Shop is currently testing a program that leads eight-inch, single density, CPM produced floppy disks containing text files and prints formatted output from them. During

this test period, procedures for handling such requests are being developed. If you are interested in such a service, contact Gene Wiley at 565-2005.

KNOW YOUR REPRESENTATIVE

The University Computing Council was established in 1981 to function as an advisory body with respect to computing policies, services, physical facilities, and allocation of resources. It is a standing Council, and reports to a Steering Committee composed of the Vice President for Fiscal Affairs, the Vice President for Academic Affairs, the Vice President for Administrative Affairs, and a Representative of the President of the Texas College of Osteopathic Medicine.

The Computing Council is concerned with both current policy matters and long-range planning. An emphasis is placed on policies and procedures to provide and maintain a balance between required computing services and the necessary funding, staff, and space commitments to meet these requirements. The equitable allocation of existing computing services is also of continuing concern.

Academic, administrative, and Computing Center representatives comprise the Computing Council, with the academic representatives serving the dual role of Council members and Academic Computing Coordinators for their respective colleges or schools. The duties of the Academic Computing Coordinators are listed below:

1. Developing, documenting, and updating school or college and departmental computing plans including current and future computing service requirements by course or project. These plans are used by the Council to determine and document University academic computing requirements.
2. Providing advice and assistance to faculty members in the use of computing services and software packages to promote effective and coordinated use of available computing services and facilities.
3. Providing advice and assistance to the Computing Center staff in the development of user guides, software documentation, and seminars.
4. Coordinating, assisting, and controlling departmental or faculty sub-allocations of resources such as terminal and on-line disk storage.
5. Coordinating, assisting, and controlling departmental or faculty acquisition of computing equipment and software (including, microcomputers and word processing systems).
6. Other duties as agreed within the Computing Council.

The administrative computing representatives are individuals who, because of their positions, are normally responsible for the development and coordination of major administrative computing information systems. Other members of the Council include the Chairpersons of the Computer Science Department and the Accounting and Information Systems Department; The Director of Computer Systems, the Manager of Academic Computing Services, and the Manager of Administrative Computing Services (all non-voting, ex-officio members); and several additional members who have expertise otherwise not available on the Council.

It is particularly important that all members of the academic community be aware of their particular representative on the Council, so that any requests or questions that would be appropriate for the Council to consider can be heard. Following is a list (as of September 16, 1982) of all Computing Council members :

Academic Computing Coordinators

Dr. Neal Tate, Chair, College of Arts and Sciences
Mr. Cengiz Capan, College of Business Administration
Dr. Tom Fairchild, School of Community Service
Dr. Paul Schlieve, College of Education and Home Economics
Dr. Hugh Kirkpatrick, Graduate School and Faculty Research
Dr. Don Cleveland, School of Library and Information Sciences
Dr. Larry Austin, School of Music
Dr. Bud Hammons, Department of Computer Sciences
Dr. Jon Clark, Department of Accounting and Information Systems
Mr. Verney Sallee, TCOM, Department of Physiology

Administrative Computing Coordinators

Academic Affairs Systems

Ms. Joneel Harris, Admissions and Records
Mr. Melvin Gouge, Student Affairs and Financial Aid
Dr. Jimmie Naugher, Office of Planning and Analysis

Fiscal Affairs Systems

Mr. George Hardin, Payroll, Accounting, Purchasing, Budgets
Mr. James Terrell, Housing, Bookstore and Other Fiscal

Administrative Affairs and Other Systems

Mr. Steve Miller, Personnel, Physical Plant, Facilities, Advancement and Others

Medical Affairs Systems

Mr. Steven R. Russell, Business Manager, TCOM Administrative Systems

Other Representatives (Non-Voting)

Mr. Richard Harris, Director of Computer Systems
Dr. Tom Madron, Manager of Academic Computing Services
Mr. Coy Hoggard, Manager of Administrative Computing Services
Dr. Jim Mackey, Hardware Specialist - Physics Department
Dr. Larry Luce, Director of Facilities Planning & Utilization
Mr. Ray McFarlane, Director of Facilities Planning & Construction
Mr. Tom Newell, Communications Manager

USING THE COMPUTER FOR RESEARCH: PART VI
By Bob Brookshire

This article is the sixth in a series of articles on computing and data analysis. The previous articles appeared in BENCHMARKS Volume 3, Numbers 2,3,4,5, and 6.

SPSS: Statistical Package for the Social Sciences

The name Statistical Package for the Social Sciences does not really describe SPSS, since it is widely used in other academic areas and in businesses. It is probably the most popular of the statistical packages. SPSS was originally developed in the 1960s by a group of political scientists at Stanford University headed by Norman Nie. Soon after, Nie and his group moved to the University of Chicago, and began to provide SPSS to other institutions. The package is now marketed from Chicago by SPSS, Inc.

SPSS Commands

SPSS commands have a fairly rigid structure. Each statement is divided into two fields, a control field and a specification field. You can think of the control field as containing a basic command, for which the statements in the specification field provide more detailed information. The control field is always located in the first 15 columns (print positions, or "spaces") of an SPSS statement, and the specification field always starts in the 16th column. Each statement cannot be over 80 characters (columns or print positions) long. Specification fields ordinarily may, however, be continued on subsequent lines, provided that the continuations also begin in the 16th column of the subsequent lines. The examples below will make the format of SPSS statements clearer.

The RUN NAME Statement

The RUN NAME statement is usually the first statement in the SPSS program control language section of your program. Its format is simple: the control field, the first 15 columns of the statement line, contains the words RUN NAME, and the specification field contains any descriptive name that you want to assign to title your SPSS output. The name in the specification field of the RUN NAME statement will appear at the top of each page of your printout. Table 1.1 shows an example of the RUN NAME statement.

TABLE 1.1

RUN NAME Statement for the Example Data

RUN NAME	ANALYSIS OF RESEARCH METHODS CLASS DATA
----------	---

Data Definition Statements

Like all statistical packages, SPSS requires that the data matrix to be analyzed be described in some detail. Several SPSS statements are required for this purpose.

THE DATA LIST STATEMENT

The DATA LIST statement gives the names and locations of the variables in the data matrix. The words DATA LIST form the control field of this SPSS statement. The specification field starts with the word FIXED/1 followed by a space, and then the name of the first variable. Following the variable name are the numbers of the columns in the data matrix in which the data for that variable are located. In parentheses following the column numbers are

1. the number of decimal places in the data values, if any, or
2. the letter A if the variable is alphanumeric, that is, it can take nonnumeric values.

If the variable takes only numeric values and has no decimal places, this may be omitted. Next comes the name of the second variable, its column numbers, etc., until all the variables in the data matrix are described.

Variable names cannot be more than eight characters long, and cannot start with numbers. For our example data set (presented in BENCHMARKS, Volume 3, Number 6, page 18), suitable variable names might be STUDENT, GENDER, TEST1, TEST2 and TEST3. Table 1.2 shows the DATA LIST statement for the example data. Note that the words DATA LIST start in the first column of the statements, and that the list of variable names starts in the 16th column of the statement. Notice, too, that when the specification field is continued on the next line, it begins in the 16th column.

TABLE 1.2

DATA LIST Statement for the Example Data

DATA LIST	FIXED/1 STUDENT 1-2 GENDER 3 (A)
	TEST1 4-5 TEST2 6-7 TEST3 8-9

Here we have told SPSS that the variable named STUDENT is located in columns 1 and 2 of the data matrix, the variable GENDER is located in column 3 and is alphanumeric, the variable TEST1 is located in columns 4 and 5, TEST2 in columns 6 and 7, and TEST3 in columns 8 and 9. None of our numeric variables, STUDENT, TEST1, TEST2 or TEST3, have decimal places.

The DATA LIST statement has other features, as well. For more details, see pp. 49-57 in the SPSS Manual. If you have a large number of variables, you may find the INPUT FORMAT and VARIABLE LIST statements to be more flexible than DATA LIST. See the SPSS Manual (Norman H. Nie, et al., SPSS: Statistical Package for the Social Sciences (New York: McGraw Hill, 1975) available at the University Store), pp. 36-38, 41-49.

The INPUT MEDIUM Statement

The INPUT MEDIUM statement tells SPSS what the storage medium for the data is. INPUT MEDIUM is the control field for this statement, and the words TAPE, DISK or CARD go in the specification field. If your data is stored on disk in the MUSIC operating system, your INPUT MEDIUM is CARD. This is because when OSJE transfers your program and data to the OS/MVT operating system, it treats them as though they were cards, since OS/MVT can only recognize card input. You will use the DISK input medium only if your data is stored on magnetic disk under the OS/MVT operating system. We will assume that our example data are stored in the MUSIC operating system. Therefore, our INPUT MEDIUM statement would look like Table 1.3.

TABLE 1.3

INPUT MEDIUM Statement for Example Data

INPUT MEDIUM	CARD
--------------	------

The N OF CASES Statement

The next bit of information we need to provide to SPSS is the number of cases (respondents, subjects, or rows in the data matrix). We do this with a statement in which the words N OF CASES appear in the control field, and the number of cases appears in the specification field. In our example data, we have 20 students. Our number of cases is therefore 20, and our N OF CASES statement looks like Table 1.4.

TABLE 1.4

N OF CASES Statement for Example Data

N OF CASES 20

The MISSING VALUES Statement

If data for some variables are missing for one or more cases, you need to inform SPSS of this. Simply leaving blanks in the data matrix is not sufficient. The reason this will not do is that SPSS will convert blanks in the data matrix to zeroes; thus, a case with a blank value for some variable will be assigned the value of zero by SPSS. This zero will then be used in the calculation of all statistics, the creation of tables, and so on.

A better procedure would be to have SPSS omit cases with missing values from the calculation of statistics. We tell the program which values are missing value codes with the MISSING VALUES statement. The word MISSING VALUES form the control field for this statement. The specification field is composed of the name of a variable, its missing value (or values) in parentheses, a second variable name, its missing value(s) in parentheses, and so on until all missing value codes have been listed. If a variable does not have a missing value code, it need not be listed. Table 1.5 shows an example of a MISSING VALUES statement.

TABLE 1.5

MISSING VALUES Statement for the Example Data

MISSING VALUES STUDENT (99) GENDER (0)

This example MISSING VALUES statement says that the missing value code for the student number variable is 99, and the missing value code for the gender variable is 0. The test score variables are not listed, so SPSS will not consider any value of these variables as a missing data code. For more information on missing values, see Benchmarks, Vol. 3, No. 4.

These are the only statements that are required in SPSS to describe the data. SPSS also provides many other interesting and useful features which can be used to describe the data matrix, however. In particular, you can define labels for the variables of up to 40 characters in length, which will be printed on your output any time the variable is used. Likewise, particular values of the variables can be labelled. These commands are described in detail in Chapter 4 of the SPSS Manual.

```

* * * * *
*
*           S P S S
*
* * * * *

```

Doing Repeated Analysis in MANOVA

Dr. Allen Jackson, Assistant Professor of Health, Physical Education and Recreation submitted the following sample setups for repeated measures analysis using SPSS MANOVA.

Basic MANOVA: 2 Dependent Variables, 2 Independent Variables

```

MANOVA          V1,V2 by A(1,2) B(1,2)/
                PRINT=SIGNIF(STEPDOWN),DISCRIM(STAN,COR)
                CELLINFO(MEANS)/
                DESIGN=A,B,A BY B
<if A by B interactions> DESIGN=A,B WITHIN A(1), B WITHIN A(2)

```

One Factor Repeated Measures (3)

```

MANOVA          V1 to V3/
                WSFACTORS=A(3)/
                WSDESIGN=A/
                PRINT=SIGNIF(BRIEF)/
                ANALYSIS (REPEATED)

```

Two Factor Repeated Measures (3* BY 3*)

```

MANOVA          V1 to V9/
                WSFACTORS=B(3),C(3)/
                WSDESIGN=B,C,B BY C/
                PRINT=SIGNIF(BRIEF)/
                ANALYSIS (REPEATED)

```

Two Factor Mixed Repeated Measures (3 by 3*)

```
MANOVA      V1 TO V3 BY A(1,3)/
            WSFACTORS=B(3)/
            WSDSIGN=B/
            PRINT=SIGNIF(BRIEF) CELLINFO(MEANS)/
            ANALYSIS(REPEATED)/
            DESIGN=A/
```

Three Factor Mixed Repetaed Measures (3 by 3 by 3*)

```
MANOVA      V1 TO V3 BY A(1,3) B(1,3)/
            WSFACTORS=C(3)/
            WSDSIGN=C/
            PRINT=SIGNIF(BRIEF) CELLINFO(MEANS)/
            ANALYSIS (REPEATED)/
            DESIGN=A,B,A BY B/
```

Three Factor Mixed Repeated Measures (3 by 3* by 3*)

```
MANOVA      V1 TO V9 BY A(1,3)/
            WSFACTORS=B(3) C(3)/
            WSDSIGN=B,C,B BY C/
            PRINT=SIGNIG(BRIEF) CELLINFO(MEANS)/
            ANALYSIS(REPEATED)/
            DESIGN=A
```

*Repeated

```
* * * * *
*                               *
*           B M D P           *
*                               *
* * * * *
```

BMDPB2 Installed

The latest release of BMDP has been installed and is now the default when executing the BIMED or BMIEDT procedures. New features of the 1982 version include:

- A powerful convenience feature which causes the program to generate repetitions of a basic form specified by the user. For example, you can now instruct the program to repeat the same transformation over a range of variables, repeat a REGRESSION paragraph over a range of dependent variables, or repeat an entire problem over a range of subsets.

- An additional feature, which allows the user to include selected sections of BMDP instructions from an external file and substitute specified parameter values for symbols in the included instructions. This "macro" capability can be used, for instance, to build a library of commonly used complex transformations. When a particular transformation is desired, the user points to the "macro" defining the transformation and supplies the variable name or names to which the transformation is to be applied.
- BMDP programs now scan all the common instructions for syntax and consistency problems. In addition, many of the programs will ensure that all program-specific instructions are scanned before terminating due to an error condition.
- The univariate and bivariate spectral analysis program, BMDP1T, is now available as part of the April 1982 release of BMDP. P1T is documented in the 1981 BMDP Manual and the 1982 User's Digest. However, a few changes have been made to the design since these were published. You can obtain these changes by stating: /PRINT NEWS. the first time you run P1T.
- Several new features have been added to BMDP4V:
 - In the BETWEEN paragraph, specification of CODES or CUTPOINTS for each factor is no longer required. The program will allow up to 10 group code values for any factor for which grouping information is not supplied.
 - In the WITHIN paragraph, category information for the within factors was formerly required. This was unnecessarily confusing, since most within designs are completely crossed and the category values did not appear with the data. To specify the number of levels of the within factors in a completely crossed design, the LEVELS sentence can be used. For instance,

```

/ WITHIN FACTORS ARE DAY,TEST.
  LEVELS ARE 5, 2.

```

describes a design where each subject receives two tests on each of five days.

- In the case of repeated measures designs with a single dependent variate, the program guessed the name of the dependent variate by using the name of the input variable associated with the first measure. This was fine as long as the user was aware that the program did this. In most cases, however, the output showed a label for the dependent variate, such as TEMP1, which made it appear as though only the first measure was being analyzed. In the current version of the program, the name of the dependent variate in a univariate repeated measures design is specified by using the DEPVAR sentence in the WITHIN paragraph. For example,

```

/ WITHIN FACTOR IS DAYS.
  LEVELS ARE 5.
  DEPVAR IS TEMPRIUR.

```

- The analysis of variance table generated by P4V is designed to accommodate a large number of possibilities, such as multivariate statistics, multivariate tests of repeated measures, reports of loss of numerical accuracy, etc. Although we consider presentation of all of these features essential, it is often desirable to have a briefer presentation of all tests for a single dependent variable. To obtain a summary table for each dependent variate in a classical ANOVA format (almost identical to P2V), state UNISUM in the ANALYSIS paragraph.

- The following corrections have been made in BMDP programs:

PROGRAM	CORRECTION
1D	All requested grouping variables are now printed, instead of the first and last only. The statistics are unaffected.
2D	Stem-and-leaf requests are produced without occasional "divide by zero" errors. This error used to be triggered when 1) data contained fewer than 3 distinct values, or 2) page width was too narrow to print stem-and-leaf output.
3D	Inclusion of a grouping variable in the list of variables to be used is permitted; it no longer causes the list to be misread. The COMPLETE option is based on the USE list of the VARIABLE paragraph, and not on the VARIABLE list of the TEST paragraph.
5D	An addressing error that occasionally occurred when attempting to plot individual groups has been corrected.
6D	Indices of variables larger than 99 are now printed correctly. The statistics option now defaults from problem to problem. Each GROUP statement in the PLOT paragraph yields a separate plot. Previously, all plots were printed in one frame if the COMMON option was used.
7D	The Bonferroni probabilities are now computed using the significance level divided by k (the number of comparisons), rather than by using a formula that includes the $K^{1/n}$ root. The Levene and Welch tests are now computed correctly, even when the data contain empty cells.

PROGRAM	CORRECTION
8D	Computational accuracy has been improved for cases where a large percentage of values is missing. In order to make the test computation more accurate, when one of the sums of squares is very small relative to the total (e.g. 10^{-5})*, only the group with the larger sum of squares is used to estimate the within variance. 10 ⁻⁵
9D	The presence of cases with USE=0. no longer results in cases being lost off the end of the data set.
4F	Tables in which one factor has a large number of levels no longer print extremely slowly. Several conditions that caused addressing errors were corrected. Division by zero no longer occurs when a marginal subtable is empty. When a table is saved and there are duplicate group names, the grouping information is now correct for re-analysis of the table.
1L	This program previously accepted up to 30 status codes although it was prepared to use only 10. It now accepts no more than 10.
2L	The default for the number of step halvings is now 5.
2M	Variable labels are now printed correctly even when a weight variable is used or the USE list is not in lexicographic order.
4M	The program no longer malfunctions when the WEIGHT variable is not in the list of variables to be used.
8M	No further computations are attempted if the number of responses or discrepancies is zero. Row and column labels are no longer printed out of phase when some cases are not be used.
KM	Standard deviations are now computed by division by (n-1) rather than by n. The program now saves the correct cluster indicator even if a cluster has zero variance. Square root error due to rounding has been eliminated.

PROGRAM	CORRECTION
2R	The name of the case weight variable is now printed correctly. The program no longer terminates with a memory addressing error if the weight variables or case labeling variables are in the list of variables to be used.
3R	Occasional errors in the handling of constraints have been corrected. The printing of more than 6 parameter estimates no longer fails.
9R	When normal probability plots are the only plots requested, the program now produces them correctly, rather than being stopped by various error conditions. Cases with zero weight are now reported as such, rather than under the wrong heading "CASES WITH GROUPING INDEX=0." Because it is too liberal, the test for goodness-of-fit of zero weight cases is no longer computed.
LR	The C. C. Brown statistics are now computed more conservatively, so that singularity is diagnosed more frequently.
2T	The handling of missing data is improved in several parts of the program. The program now analyzes only the series from the first non-missing data value up to and not including the next missing value. No warning is given. The acceptable abbreviations for FITER, PCORR, and METHOD in the ESTIMATE paragraph were shortened. The program now checks to make sure that the new variable specified in the DIFF paragraph is not an existing variable name. When there are a small number of points in the series, the limitation on the DFORDER parameter is now correctly computed. When an unrecognized paragraph name is specified, P2T now gives an error message rather than terminating.
2V	An undefined variable no longer causes an excessive space request for array X0 to appear as a diagnostic message. When running multiple problems, the grand mean reported for the second and subsequent problems is now correct.

PROGRAM	CORRECTION
4V	Replications of WITHIN cells are now recognized as such rather than being treated as separate cells. The MARGINALS sentence in the PRINT paragraph is no longer mistakenly restricted to a single value.

If you want, for some reason, to use the 1981 version of BMDP, use the statements: EXEC B1MED81 or EXEC B1MEDT81. If you encounter any problems with BMDP, contact Bob Brookshire at 565-2324.

```

* * * * *
*           *
*       S   A   S       *
*           *
* * * * *

```

New Release of SAS Installed

The latest release of SAS has been installed and is the procedure you get when you // EXEC SAS. Release 79.6 will serve as a bridge between the SAS 79 series and their next major version, SAS 82. The documentation for SAS 79.6 is still the SAS User's Guide: 1979 Edition, the SAS/ETS User's Guide: 1980 Edition, and the SAS Supplemental Library User's Guide: 1980 Edition, however, the SAS User's Guide is out of print. It will be necessary, therefore, to buy the manuals that go with SAS 82, even though we do not have that version installed on our system yet. This will not be a major problem, because the 1982 manuals have all the information in them that is needed to use SAS 79.6. They also, however, have information on features that are not currently available. If you attempt to access a procedure or function that is not included in SAS 79.6 you will get an error message that states, in effect, that it is not available.

The new SAS manuals, available in the University Store, are: SAS User's Guide: Basics and SAS User's Guide: Statistics. It is not absolutely necessary to buy both manuals, however, if you want to do anything beyond descriptive statistics, the Statistics Guide must be purchased.

New Features of SAS 79.6

New System Options

Three new system options have been added to this version of SAS:

- BLDTABLE/NOBLDTABLE
- NOQUOTE/DQUOTE
- PROCSIZE=

Additionally, this version of SAS was installed with the system option S=0. This means that the first line of a SAS job, usually the DATA; line is checked for numbers in columns 73-80. Because of this, nothing with a number in column 73 should be included on the first line of the job, as SAS would assume all subsequent lines, including the data, contained sequence numbers in those columns, and would ignore them. Of course, if the entire job actually did contain sequence numbers this would be the desired result.

For a complete description of the SAS system options, execute PROC OPTIONS.

SAS Data Libraries*

Beginning with release 79.6 of SAS, SAS data libraries may be specified in the job control as DD DUMMY; any DCB specification is ignored and is not altered.

For input data sets, a _NULL_ data set is assumed. For output data sets, no data are written. In some cases this feature can replace the SAS system option OBS=0.

This feature allows you to control the SAS data libraries read or written in a SAS job by changing only the job control. For example, suppose you run a weekly production job which produces a summary and a detailed report. Normally, you want only the summary printed; the detailed report is printed only if you need an audit trail. By specifying the DETAIL SAS data library as DD DUMMY, you can produce a summary report only.

```
//SUMMARY JOB (Lenn-nnnn), 'MY NAME'  
//SUMMARY DD DSN=UESRn.Dnnnn.Pnnnn.MY.SUMMARY.DATA, DISP=OLD  
* //DETAIL DD SUMMARY DUMMY  
DATA SUMMARY.MAY DETAIL.MAY;  
statements to produce reports  
PROC PRINT DATA=SUMMARY.MAY;  
PROC PRINT DATA=DETAIL.MAY;
```

Beginning with 79.6, OS users <that's us> must allocate SAS disk data libraries <data bases> with DCB characteristics DSORG=DA, RECFM=U. This is to prevent the possibility of destroying a non-SAS data set by incorrectly referencing the data set as if it were a SAS data library, and to prevent SAS data libraries from being copied or moved incorrectly.

Enhancements to GLM/ANOVA

Several new multiple comparison methods are available in PROC GLM and PROC ANOVA. Specify the options in the MEANS statement:

- BON Bonferroni t tests
- LSD Fisher's least-significant difference test,
 equivalent to multiple t tests
- SCHEFFE Scheffe's multiple-comparison procedure
- SNK Student-Neuman-Keuls multiple range test
- SMM studentized maximum modulus test
- TUKEY Tukey's studentized range test (HSD)

The ALPHA= option may take any value between 0 and 1 for all the options listed above.

Experimental procedures*

Three experimental procedures are available with SAS 79.6: TPRINT, TEDITOR, and TARIMA. TPRINT is an experimental pre-release version of PROC PRINT that allows you to specify variable labels for column headings. The additional options needed to produce variable labeling are described below.

LABEL

The LABEL option prints variable labels, if any, instead of variable names. A LABEL statement is used to give variables their labels. The LABEL statement may be used in the PROC TPRINT step or in the DATA step that builds the data set you want to print. (If the LABEL option is not specified, TPRINT uses only variable names).

SPLIT =character

The SPLIT=option specifies a character that defines where to split the label. For example, SPLIT='#' tells SAS that the label 'THIS IS#A LABEL' is to be printed

```
THIS IS  
A LABEL
```

If SPLIT= is used, the LABEL option is implied. If the SPLIT= option is not used, TPRINT splits the label as close as it can to the center of the label (excluding trailing blanks). If you do not want the label split at all, add the split character to the label after the last non-blank character.

If you want the procedure to omit the column header for variable X, use the statements

```
PROC TPRINT SPLIT='*';  
  LABEL X='*';
```

Since the procedure never prints labels vertically, using the LABEL option with long variable labels may increase the number of pages needed to print the data set considerably.

The TEDITOR procedure is an experimental pre-release version of the current EDITOR procedure with these new features:

- Additional operators for FIND command: $\neg$ =, >, >=, <, <= For example, the command: FIND 1,5 X=2 Y<6 Z $\neg$ =7; searches observations 1 through 5 for the first observation where X is equal to 2, Y is less than 6, and Z is not equal to 7.
- LOCATE and NAME commands.
- SEARCH and STRING commands.
- DUP command.
- Ability to easily repeat FIND, SEARCH, and LOCATE commands. This feature of PROC TEDITOR has no practical use for batch users of SAS <that's us!>.
- Handles special missing values, date, time, and hex literals, but only if no informat is assigned to the variable. For example, if no informat is assigned to the variable BIRTHDAY, the statements

```
NAME BIRTHDAY;  
LOCATE 1, LAST '10JUN48'D;
```

search all observations for the value 10JUN48.

- Adds a semicolon to the end of LISTed observations. Again, this procedure is of no help to batch users.
- The END command has two additional aliases, QUIT and Q.

The new procedure is designed to be more compatible with SAS/FSP <a SAS product we do not currently have, but plan to get> than the old PROC EDITOR.

LOCATE and NAME commands

NAME variable;

The NAME command is used in conjunction with a LOCATE command. It gives the name of a variable to be used in a LOCATE command for locating variables. The LOCATE command specifies the range of observations in which to search and the value of the variable you want to locate in that range. For example, the commands:

```
NAME X;  
LOCATE 1,10 2.35;
```

tell the procedure to search the values of X in the first through the tenth observations for a value 2.35. Some examples of LOCATE commands;

```
LOCATE 1, LAST ABC;
```

searches all observations for the character value ABC. The keyword LAST may be used when you want to search the entire data set and you do not know the total number of observations.

```
LOCATE 6,20 'X,Y,Z';
```

searches observations six through twenty for the character value 'X,Y,Z.' Note that if the character value contains special characters, the value must be enclosed in apostrophes(').

```
LOCATE 20,30 10;
```

searches observations twenty through thirty for a value 10. The variable named in the NAME command prior to this LOCATE command could be either a numeric or a character variable.

```
LOCATE: 1, LAST SMITH;
```

searches all observations for a value beginning with the letters SMITH. A colon (:) after the command name tells SAS to search for leading characters only. Note that this LOCATE command would find names such as SMITHFIELD or SMITHDALE. To find only last names of SMITH, enter LOCATE: 1, LAST 'SMITH,';

STRING and SEARCH commands STRING variable;
SEARCH string;

The STRING command specifies a list of character variables used by the SEARCH command. The SEARCH command lists one or more strings you want to find in a range of observations. For example, the commands

```
STRING NAME SPOUSE;  
SEARCH 1,100 SMITH JONES;
```

tell SAS to search the variables NAME and SPOUSE in the first 100 observations for the string SMITH or the string JONES.

A string can be found anywhere within a value, not just at the beginning.

Up to ten strings may be specified in a SEARCH command; the total length of all strings given may not exceed 200.

If a string contains special characters, it must be enclosed in apostrophes.

In the example above, SAS finds only observations with both strings 'SMITH' and 'JONES'. To find either 'SMITH' or 'JONES' in one of the variable fields NAME or SPOUSE, enter

```
STRING NAME SPOUSE;  
SEARCH @ 1,100 SMITH JONES;
```

DUP command

DUP observation;

The DUP command is used in TEDITOR to duplicate observations in a specified range and write them on the data set. For example,

```
DUP 1,3;
```

tells the procedure to copy the first, second, and third observations to the end of the data set.

PROC TAEIHA is a test version of the current ARIMA procedure in SAS/ETS. It performs subset models and transfer function models.

A New Procedure: PROC APPEND*

The APPEND procedure copies observations from one SAS data set onto the end of another SAS data set.

A large data set is often used to accumulate observations. The number of observations being added at any one time may be small, but SAS must read all observations when concatenating the smaller to the larger using these SAS statements:

```
DATA LARGER; SET LARGER SMALLER;
```

This DATA step takes a lot of computer time if the LARGER data set is very large. However, the APPEND procedure copies the observations from the SMALLER data set onto the end of the LARGER data set without reading through observations in LARGER.

The statement used with PROC APPEND is: PROC APPEND options; No other statements are used with the procedure. These options may appear in the PROC APPEND statement:

ORIG =data_set

ORIG= gives the name of the original data set onto which the new observations will be added. This option must appear. It can also be referred to as BASE=, MASTER=, or OUT=. The data set does not have to already exist. (See the NEW option below.)

DATA

=data_set

DATA= gives the name of the data set whose observations will be copied onto ORIG. If DATA= is omitted, SAS copies the most recently created SAS data set.

NEW

The NEW option tells SAS that instead of adding observations onto the end of ORIG, the observations should replace the ORIG data set entirely. The NEW option is assumed if the ORIG data set does not already exist.

Common Characteristics of the ORIG and DATA Data Sets

If there are differences between the variables in ORIG and NEW, variables in NEW are either truncated or padded to match the lengths of corresponding variables in ORIG. Variables not common to both are dropped if in ORIG and given missing values if in NEW.

If variables in the data sets are the same (that is, have the same positions and the same lengths), the procedure runs faster. Run PROC CONTENTS on both data sets to see that the two data sets are of identical structure.

Output from PROC APPEND

The only printed output that APPEND produces is a message on the log indicating the number of new observations added to the ORIG data set. The procedure also prints a message if ORIG is being created as a new data set.

A NOTE on Performance

PROC APPEND operates on the ORIG data set in "update" mode, in similar fashion to PROC EDITOR.

Examples

In this example, NEW contains the day's accounting data. PROC APPEND adds NEW to the journal data set OUT.JOURNAL.

```
DATA NEW;  
  INPUT ACCTNUM AMOUNT SALESREP; CARDS;  
  data lines  
PROC APPEND ORIG=OUT.JOURNAL DATA=NEW;
```

The example below appends data set TEMP to itself.

```
DATA TEMP;  
  statements to create data set  
PROC APPEND ORIG=TEMP;
```

A copy of data set X is stored in data set Y.

```
PROC DELETE DATA=Y;  
PROC APPEND OUT=Y DATA=X;
```

SAS/ETS: New Features

- In PROC AUTOREG, the ITER option in the MODEL statement performs iterative estimation of autoregressive and regression parameters.
- PROC TIMEPLCT, a test version of the new procedure, plots time-series data.
- Three new time-series functions are available in PROC MATRIX:
 1. ARMACOV autocovariance for an ARMA model
 2. ARMALIK log likelihood for an ARMA model
 3. MRATIO matrix impulse response function

SAS Sample Library

An extensive set of SAS sample programs have been installed along with the new release, and are available for execution. These sample programs

exercise many lesser-known SAS facilities. Many classical statistical problems are illustrated by the sample programs, which makes them excellent tools for instructional purposes. The following setup illustrates how to run a program from the sample library. To get an index of sample programs, simply use this setup, substituting the word INDEX for "name of program."

```
//SAM JOB (nnnn-nnnn,1,20),'MY NAME'
//*
/* JOB TO RUN SOME SAMPLE PROGRAMS
/*
//SHOWOFF EXEC SAS
//SAMPLE DD DSN=KSRCH.D0683.P8608.SAS.SAMPLE,DISP=SHR
//SYSIN DD *
%INCLUDE SAMPLE(name of program);
%INCLUDE SAMPLE(name of next program);
%INCLUDE SAMPLE(name of last program);
/*
//
```

SAS PROC BMDP

SAS BMDP is a procedure that allows you to run BMDP programs without exiting from SAS. Below is an example of using PROC BMDP:

```
//SASBMDP JOB (nnnn-nnnn,:10,1),'MY NAME'
// EXEC SASBMDP
//DATA DD DSN=USERn.Dnnnn.Pnnnn.SASFILE,DISP=SHR,
// UNIT=SYSDA,VOL=SER=ACAD02
//SYSIN DD *
PROC BMDP PROG=BMDPAM DATA=DATA.SASFILE; PARMCARDS;
/PROBLEM TITLE IS 'DESCRIPTION OF MISSING DATA IN SAS SAVEFILE'..
/INPUT UNIT IS 3.
CODE IS SASFILE.
/ESTIMATE METHOD IS REGR.
/END
/*
```

NOTE that unlike other SAS procedures, you must execute SASBMDP in order to access PROC BMDP.

*Adapted from SAS Communications Vol. VII Number 4, Spring 1982.

PROC COMPUTAB: SAS's Answer to VISICALC

Those in the know say that PROC COMPUTAB, described in the SAS/ETS User's Guide, provides the same sort of functions available with the popular microcomputer software package VISICALC. It is ideal for dealing with spread-sheet applications, as new columns and rows can be defined as functions, percentages can be generated, etc.

The tables generated by PROC COMPUTAB are more customized than those provided by PROC MEANS, PROC SUMMARY, or PROC PRINT. It is also easier

to generate customized reports using COMPUTAB than it is using PUF statements, etc., in a data step.

```
* * * * *
*           *
*      M U S I C      *
*           *
* * * * *
```

STATPAK Manuals Available

The STATPAK User's Guide and the STATPAK Reference Manual are now available in the University Store at a cost of \$2.15 and \$5.35 respectively. These manuals should provide additional help in using the MUSIC STATPAK facility.

ASCII Terminals Under VM

All ASCII terminals that were hardwired to MUSIC have been brought up under the VM/370 operating system. These terminals are readily identifiable, because, when turned on, they display the message "VM/370 on Line" (you may have to depress the <RETURN> key to get the message).

The Virtual Machine 370 operating system is a program that supervises other operating systems, and allows communication between those operating systems under its supervision. A terminal attached to VM, therefore, allows access to the three other operating systems currently running on the AS/5000: MUSIC, OS/MVT, and CMS. Additionally, these terminals have the ability to emulate 3270 type terminals when connected to the 3270 Simulator (SIM3270). A reference card describing how to gain access to the various operating systems through VM/370 is available at the Media Library, BA RJE, Computing Center RJE, or from the Computing Center receptionist in ISB 114.

It should be noted that each of the systems cannot be accessed without the proper ID-codes and passwords. Should you try to use a system that you do not have access to, problems may arise. If you dial MVTDSX during a time when COM-DELETE is not up, the only way to return to VM is to have the operator disconnect the line. This requires that you provide the port number (address) to which you are attached. This information is provided in a message which is received when the system is dialed, and has the following format:

DIALED TO COM-DELETE nnn where 'nnn' is the port address. If you have cleared the screen on your terminal, this information will no longer be available to you, and the operator will have to find the appropriate line to detach.

If you have dialed MVTDSX during a time when COM-DELETE is up, you must issue the command: *ULOG OFF to get out of COMPLETE. If you have previ-

ously dialed SIM3270 and have not successfully logged on to COM-LETE, you must log on to COM-LETE using the command: *ULOG ON,HELPER with the password HELP You can then enter the command: *RSET to return to VM. If, by some chance, you have gotten in to CICS, the command: RSET will get you out. If you have unsuccessfully attempted to log-on to CMS, the command: LOGOFF will get you back to the VM logo.

If you wish to gain access to a system that you are not currently authorized to use, you may obtain a "NTSU, Computing Center Service Request Form" from the Computing Center receptionist.

ID-Code Renewal

All MUSIC ID-Codes expired on September 30. If you have not renewed your ID-Code, the time has been set to zero and you will not be able to log-on. These zeroed out ID's will be kept on the system until the end of October, at which time they will be permanently purged. To renew your ID, an "NTSU Computing Center Request Form," available from the Computing Center, must be filled out.

Disk Space Limited!

Once again it is necessary to remind all MUSIC users that there is a finite amount of disk space available. This means that YOU must manage your MUSIC files in an intelligent manner. If you have files that are no longer useful, PURGE them. If you have files that are seldom used, dump them to tape using the #EXCHANGE utility and then PURGE them from your ID. It is going to take an act of GOD to get more disk space this semester!

Microcomputer Software on MUSIC

The following article is being reprinted from the May 1982 issue of BENCHMARKS, Volume 3, Number 4.

INTRODUCTION

Because of the rapid growth in the use of microcomputers at NTSU we are providing a library of contributed programs and related documentation for your use. The library will continue to be built through the contributions of users of MUSIC--you may be among those contributors. If you have a program to contribute, please send a message, via MAIL, to SYSTEM, indicating your USER ID, and the name of the program. Any program you contribute should be your own or already in the public domain. It should not be one you have purchased from a Computer Store or other commercial enterprise. Likewise, these programs are available for your personal use, not for piracy and resale.

In a similar vein, any programs or other files on MICROLIB are made available for their educational and instructional value as well as the enjoyment of computer users at North Texas State University. Any programs or data are not warranted in any way by NTSU, by The Computing Center, or by Academic Computing. If the contributed programs do not work, or if help and assistance is needed, please call the contributor, not Academic Computing. Academic Computing will be happy to assist you in using MICROLIB, but cannot take responsibility for contributions made by others.

CONTRIBUTIONS

While this service is conceived primarily for microcomputer users, if you have written a program for MUSIC you would like to share with others, please let us know. In order to use this service effectively you must have terminal software capable of "uploading" and "downloading" files to and from MUSIC. If you have a BASIC program on an APPLE you would like to share, for example, you would first need to save a copy of your program as an ASCII file on your APPLE system. You would then enter your terminal program and dial MUSIC. At that point you would need to upload the program to your USER ID. Once that is accomplished, you would then send a MAIL message to SYSTEM and we will take care of the rest.

If you wish to contribute a program, please make sure that you have placed comment statements at the beginning of the program in the following format:

- Program name and purpose
- Contributor's Name
- Contributor's Address and Phone Number
- Date Program was written (and/or last update)
- Date of Contribution
- Any operational notes required
- Bibliographic citation if the program has been published

In the case of compiled, machine language files, there is only one way such programs should be contributed. They should be in Intel's HEX format (either as the byproduct of compilation, usually on a CP/M system, or by UNLOADing a CP/M COM file). If you are using a TRS-80 with the ST80III communications program, ST80III comes with utilities to change CMD (the TRS-80 equivalent of the COM file) files to the Intel HEX format. In this event, you should also provide a DOC file with the information described above for each program contributed. On CP/M systems the HEX files can be turned back into COM files with the LOAD utility. COM files can be turned into HEX files with the UNLOAD utility (LOAD is a standard CP/M program, UNLOAD is from the CP/M Users Group Library and is included in this library).

Each program in the library has a three level name: `system.filename.filetype` "System" pertains to the type of system for which the program was written: CPM, TRS80, APPLE, MUSIC, etc. "File-

name" is a simple name for the program of eight characters or less, beginning with an alphabetic character. "Filetype" is a three character sequence indicating the kind of file (BAS, for BASIC programs; DOC for Documentation; HEX, for HEX files; etc.). In order to determine whether there are any programs or files of interest to you, you can simply issue the MUSIC LIST command, as follows: LIST MICROLIB.system where "system" is a term as described above.

To get a list of the various systems for which some items may be available, issue the following MUSIC command: LIST MICROLIB.SYSTEM <exactly those words> This command will provide a correct list of applicable "systems". As of this writing the list is more a "wish" list for contributors than a list of real resources, although there are some files already on MICROLIB.

NOTES ON UPLOADING AND DOWNLOADING

In order to use MICROLIB you must be using a microcomputer as a "smart" terminal. That is, your communications equipment must have the capability for "uploading" and "downloading" files. This point has already been made, but cannot be forgotten. The general procedure for uploading from your micro to MUSIC is to load a file into your storage buffer on your micro. Go back to communications and invoke the MUSIC EDITOR using EDIT or TEDIT commands. Wait for the word INPUT to appear, along with the first prompt (?). Use your microcomputer terminal program to start the sending of data. When all data have been sent, go back to communications and press your <ENTER> or <CARRIAGE RETURN> key. This will terminate the INPUT function and return you to EDIT. Save your file. In general, you will find it difficult to upload a program or file to MUSIC when any line exceeds 80 characters in length. You may have to edit some files before transmission.

In order to download a file from MICROLIB decide which file you wish to have. ReLOG to your User ID using a terminal class which will not generate the MORE message: /ID USERID;LA36. Issue the MUSIC command "LIST filename" but do not press <ENTER> or <CARRIAGE RETURN>. Open a memory buffer on your microcomputer. Go back to communications and press <ENTER> or <CARRIAGE RETURN>. MUSIC will now start listing the file specified by "filename." When the entire file has been listed, close your memory buffer and save the file to disk (or possibly tape) on your micro. The file you have saved on your micro will have, at the beginning and end, some extraneous lines (usually of MUSIC messages). You must now use a text editor to take those extra lines out. If you are using a TRS-80 with ST80III you can, instead, issue the command to MUSIC, ST80DOWNLOAD. A downloading program which asks for filename and record length will then open the micro's memory buffer automatically and close it automatically. As we receive requests, and as it is possible, we will provide similar programs for other microcomputer communications programs and systems.

WHEN ALL ELSE FAILS -

If you need further help in using the MUSIC MICROCOMPUTER SOFTWARE LIBRARY, use MAIL to send inquiries to SYSTEM or call Academic Computing at 817-565-2324. On-line help may be obtained by entering: LIST MICROLIB.DOC

MUSIC Usage Statistics 1981-1982

Number of ID-Codes by Semester

Time Period	Faculty & Research Individual ID's	Business Administration Class ID's	Computer Science Department Class ID's	Other Departments Class ID's
* Fall 81	453	504	958	86
Spr. 82	550	624	678	173
Sum. 82	553	263	273	37

Average Number of ID-Codes, Files, and Amount of Space Used*

Month	Year	Codes	Files	Space (K)
September	1981	489.6	6896	62370.8
October	1981	804	9827.3	71897.3
November	1981	987.25	12340.5	83963.5
December	1981	1077.8	13382.4	91493.6
January	1982	613.25	9314	75342.5
February	1982	863	10636.5	80551.5
March	1982	1054.5	12818.25	90955.5
April	1982	1177.7	15421.5	103323
May	1982	1223.25	16543.25	110710.5
June	1982	738.2	11588.8	93138.81
July	1982	753.2	11788.5	96108
August	1982	723.2	12135	99900.8
September	1982	1018	13309.75	99992.5

*Note the fluctuations at the beginning of each semester, after ID's have been purged from the system.

```

* * * * *
*           *
*   H P - 2 0 0 0   *
*           *
* * * * *

```

HP Notes

- Those of you who have been using IDA might be interested to learn that a book on using IDA on the HP is now available at the University Store. The book is Conversational Statistics by Harry V. Roberts. It costs around \$17.00. If you would to try IDA (Interactive Data Analysis) for the first time, enter EXE-\$IDA and respond "YES" when it asks you if you would like instructions.
- It is possible to "upload" and "download" your micro files to and from the HP if your communications software has the required capabilities. If you are interested in HP-to-micro communications, contact Jeff Brooks at the Computing Center (565-3885).

HP 2000 USAGE STATISTICS FOR '81-'82 FISCAL YEAR

Terminal Hours:	Fall '81	8696.0
-----	Spring '82	11900.8
	SummerI '82	2277.6
	SummerII '82	2970.3

	total	25844.7

Maximum Number of I.D. Codes: 2036 (Fall '81)

Minimum Number of I.D. Codes: 668 (SummerII '82)

NTSU COMPUTING CENTER GENERAL USAGE EQUIPMENT LIST
(BY LOCATION)

COMPUTING CENTER COMPUTER ROOM:

- 1 National Advanced Systems, Model AS/5000 Computer System with 4 megabytes of main memory.
- 8 NAS 7350 Disk Storage Units and Control (2.5 gigabytes)
- 8 NAS 7330-11 Disk Storage Units and Control (1.6 billion bytes)
- 4 NAS 7420 Tape Drives and Control (320-1250 KBS, <3 1600-6250 BPI, 1 800-1600 BPI>)
- 1 IBM 1403-N1 printer (1100 lpm)
- 1 IBM 2540-1 Card Reader/Punch (1000/300 cpm)
- 1 IBM 2821 Control Unit for 1403/2540
- 3 IBM 3272-2 Terminal Control Units (64 synchronous lines)
- 4 IBM 3277-2 CRT Terminals
- 1 Memorex 1270 Terminal Control Unit (16 synchronous and 80 asynchronous lines)
- 1 DIAELC 1620 Timesharing Terminal (300 Baud)

- 1 Hewlett Packard 2000 Access, Model 40, Timesharing Computer System
 - 1 HP 2108A M/20 System Processor (64K)
 - 1 HP 2112A M/30 Communications Processor (64K)
 - 1 DEC LA36 Console Terminal (300 Baud)
 - 2 HP 7905A Disk Storage Units (29.5 million bytes)
 - 1 HP 12972A Tape Drive (1600 BPI, 72 KBS)
 - 1 HP 2748B Paper Tape Reader (8 level, 500 cps)
 - 2 HP 12920b Asynchronous Multiplexors (32 RS-232 Terminal Interface Ports)
 - 1 GTE T113 BDS Data Station (12 dial-up phone lines, 110-300 Baud)

COMPUTING CENTER BJE:

- 1 Unitech UT-2 Remote Job Entry Processor <ISB 134A>
 - 1 Documentation M400L Card Reader (600 cpm)
 - 1 Unitech CT-1210 Line Printer (1000 lpm)
 - 1 Unitech KB-1 CRT System Console

COMPUTING CENTER STUDENT KEYPUNCH ROOM:

- 15 IBM 029-A22 Card Punches
- 1 IBM 059-2 Card Verifier
- 2 LSI ADM3A Timesharing Terminals (1200 Baud, direct connect to HP)
- 1 Alanthus 8400 Cassette I/O Unit
- 1 DECII (LA36) Timesharing Terminal (dial-up)
- 2 MIMC CRT Timesharing Terminals (1 dial-up)

COMPUTING CENTER RECEPTION AREA:

- 1 IBM 1230 Optical Mark Reader/Scorer
- 1 IBM 534 Card Punch Attachment for 1230
- 3 TI SILENT 700 Terminals (on check-out basis for faculty only)

MUSIC BUILDING:

- 1 DECII (LA36) Timesharing Terminal (300 Baud dial-up) <MU 1007>
- 9 LSI ADM3A Timesharing Terminals <MU 1007 MUSIC LAB - by appointment>
- 3 MIME CRT Timesharing Terminals (dial-up) <MU 2011>
- 1 DECII (LA36) Timesharing Terminal (dial-up) <MU 2011>

BUSINESS ADMINISTRATION BUILDING:

- 1 Unitech UT-2 Remote Job Entry Processor <BA 152>
 - 1 Documation M400L Card Reader (600 cpm)
 - 2 Unitech 6444 Line Printers (600 lpm)
 - 1 Unitech KB-1 CRT System Console
- 5 CRT Timesharing Terminals (dial-up) <BA 153>
- 6 DECII (LA36) Timesharing Terminals <BA 153>
- 16 CRT (MIME) Timesharing Terminals <BA 153>

GENERAL ACADEMIC BUILDING:

- 3 DECIII (LA120) Timesharing Terminals (direct connect to MUSIC) <GAB 333>
- 1 DECII (LA36) Timesharing Terminals (dial-up) <GAB 333>
- 12 CRT Timesharing Terminals (direct connect to VM/370) <Media Library>

PHYSICS BUILDING:

- 1 IBM 029-A22 Card Punch
- 1 DEC LA36 Timesharing Terminal (300 Baud) <FM 310>
- 1 Calcomp 663 Plotter <E13>
- 1 LSI ADM3A Timesharing Terminal (Direct connect to VM/370 - dial-up to HP) <B13>

WOOTEN HALL:

- 1 CRT (ACTV) Timesharing Terminal (dial-up) <WH 131>
- 1 DECII (LA36) Timesharing Terminal (dial-up) <WH 131>
- 2 DECII (LA36) Timesharing Terminals (dial-up) <WH 173>
- 2 CRT (MIME) Timesharing Terminals (dial-up) <WH 173>

SOFTWARE - WHAT WE HAVE AND WHAT WE SUPPORT

The following table is a "Software Support Key" designed to inform users of the level of support the Computing Center offers for maintenance, documentation, and consultation with regard to the software available on the AS/5000 and the HP-2000. The tables that follow this key table refer to the software available on each machine.

Software Support Key**

TYPE	FULL (F)	PARTIAL (P)	Minimal (M)	None (N)
Maintenance (MT)	a. New versions with major improvements installed as soon as possible b. Bugs fixed by CCIIR staff or reported to vendor as soon as possible. c. Vendor committed to fixing bugs.	a. New versions with major improvements installed as staff is available. b. Known errors may exist. c. Vendor may not be committed to fixing bugs.	a. New versions if available installed if no major modifications are required. b. Known errors may remain. c. No guarantee of support from vendor	No Maintenance
Documentation (D)	Full documentation from vendor or local source.	Documentation from vendor or local source may be inaccurate or incomplete.	Local documentation only on access. Other documentation may be available.	No new documentation.
Consultation (C)	Consultants familiar with use and problems of software.	Consultants have information on access, but more than that depends upon available staff.	Consultants have information on access only	No consulting help

**Letters in parenthesis denote abbreviations for each category.

AS/5000 SOFTWARE

Operating System	Processor/ Library	Support (MT) (D) (C)			Source	Comment
OS/MVT (Release 21.8F Operating System)	HASP	F	F	F	IBM/NAS	Version 4.1 (spooling processor)
	CICS	F	F	F	IBM	Version 2.3 (Teleprocessing Monitor)
	FORTTRAN IV	P	F	F	IBM	Version G
	FORTTRAN IV	P	F	F	IBM	Version H
	FORTTRAN IV	F	F	F	IBM	Version G1
	WATFIV	P	P	F	University of Waterloo	In-core FORTTRAN Lang. Translator
	COBOL	P	F	F	IBM	ANS Version 2
	COBOL	P	F	F	IBM	ANS Version 4
	WATBOL	F	F	F	University of Waterloo	In-core ANS COBOL Language Translator
	PL/I	P	F	F	IBM	Version F
	PL/I	F	F	F	IBM	Optimizing Compiler
	PL/C	P	F	F	Cornell University	In-core PL/I Lang. Trans.
	FORMAC	N	F	M	IBM Employee Contributed	PL/I F Superset
	ASSEMBLER	P	F	F	IBM	Version F
	ALGOL	P	F	P	IBM	Version F
	RPG	P	F	P	IBM	360 Version
	GPSS	P	F	N	IBM	360 Version
	SIMSCRIPT	F	F	M	Consolidated Analysis	Version 11.5
	SNOBOL	P	F	P	Bell Telephone	Version 4
	SPITBOL	P	F	P	Illinois Inst. of Technology	In-core SNOBOL4 Lang. Translator

Operating System	Processor/ Library	Support (MT) (D) (C)			Source	Comment
OS/MVT	NTSU STATPAK	F	F	F	NTSU	
	SPSS	F	F	F	SPSS Inc.	Release 9
	SAS	F	F	F	SAS Institute Inc	Release 79.6
	ETS	F	F	F	SAS institute Inc.	SAS Econo- metric & Time Series Library
	OSIRIS	N	F	P	University of Michigan	Version III
	BND	N	P	P	UCIA	
	Misc. Prog. Library	F	F	F	NTSU	
	Misc. Sub- Routines	F	F	F	NTSU	
	OS Utilities	P	F	F	IBM	
	CASOR1	F	F	F	Computer Assoc.	
	User Library	N	N	N	NTSU users	Programs which could benefit others.
	Scientific Library	N	F	P	IBM	Use is not reccmmended.
	FORNA1	P	F	M	IBM/Stanford	Release 5
	ADABAS				Software AG	
	NATURAL				Software AG	
	COM-LETE				Software AG	
	Strong- Campbell Interest Inventory	M	P	M	Brigham Young Univ.	
	LINEACK	M	P	M	Argonne National Laboratory	
	Multi- Dimensional Scaling Package	M	P	P	Bell Telephone Laboratories	
	PRIME				University of Texas	
	BMDP	F	F	F	UCLA	1982 Version

Operating System	Processor/ Library	Support (M) (D) (C)			Source	Comment
OS/MVT	VM-CMS Interface					
	WATERLOO SCRIPT				University of Waterloo	
	WATFIV-S					
MUSIC (Mc Gill)	FORTTRAN IV	F	F	F	IBM	Version G1
University System for	COBCL	P	F	F	IBM	ANS Version 4
Inter-active Computing- (Version V)	WATBOL	F	F	F	University of Waterloo	In-core ANS COBOL Language Translator
	PASCAL	P	P	P	McGill/Stanford	
	VS ASSEMBLER	F	F	F	McGill/IBM	OS/VS
	VS BASIC	F	F	F	McGill/IBM	
	APL	P	P	P	McGill/IBM	APL S/360 III
	STATPAK	F	F	F	McGill	
	IIS	F	F	F	McGill/IBM	CAI Package
	TEACH	M	F	P	Maricopa Com Col	CAI Package
	TESTER	M	F	P	Maricopa Com Col	Test Generator
	MAIL	M	F	F	Maricopa Com Col	Electronic Mailing System
	CALCOMP Plotting Routines	M	F	P	CALCOMP	
	Response Time Monitor					
	PLI	F	F	F	McGill/IBM	Optimizing Compiler
VM/370 (Virtual Machine Operating System)	VM/SP					

Operating System	Processor/ Library	Support (MT) (D) (C)	Source	Comment
CNS	Query By Example			
	COBOL			ANS Version 4
	WATERLOO SCRIPT		University of Waterloo	

There's not a prayer!

INDEX TO PAST ISSUES

In order to utilize BENCHMARKS to its fullest capacity and avoid redundancies, an index of previous issues containing information considered still pertinent to the NTSU Computing Center is included in each issue.

<u>SUBJECT</u>	<u>VOL/NO</u>	<u>MONTH/YEAR</u>	<u>PG</u>
1. ACADEMIC COMPUTING IN GENERAL BENCHMARKS Revisited: Important Items From the Past	3/①6	July/Aug/82	22
2. HARDWARE Terminal Acquisitions for New Computing Facilities	3/6	July/Aug/82	5
Microcomputer Policy Reminder	3/6	July/Aug/82	5
3. SOFTWARE A) General TUIT- A System for the Analysis of Literary and Textual Materials	3/6	July/Aug/82	6
B) MUSIC Change That Password	3/6	July/Aug/82	20
G) HP-2000 Editing BASIC Programs	3/①6	July/Aug/82	21

GET A "SUBSCRIPTION" TO BENCHMARKS

BENCHMARKS is a vital link between the NTSU Computing Center and the users of our facilities.

It is important for all users of the computing facilities to maintain a file of these newsletters because they contain materials which will periodically update existing documents as well as information and suggestions on uses of OS-MVT, MUSIC, the HP-2000, and other resources available to NTSU students and faculty.

To facilitate the dispersal of BENCHMARKS,

*** FREE ***

subscriptions

are now available.

To receive yours, send the following information to us either by "snail mail" (the post office or campus mail) or electronically, through the MAIL facility on MUSIC.

Name _____

Mailing Address _____

PLEASE GIVE A CAMPUS ADDRESS (NOT BOX) IF POSSIBLE - It's Cheaper !

PLEASE RETURN TO:

Academic Computing Services
The Computing Center
NT Box 13495
North Texas State University
Denton, TX 76203